CLAIMS

Listing of Claims:

1. (Previously Presented) A framework for isolating a business component from specific implementations of a datastore, comprising:
   (a) a database wrapper in communication with a business component;
   (b) a domain object factory in communication with the database wrapper;
   (c) a domain object in communication with the domain object factory; and
   (d) a datastore in communication with the domain object, wherein the database wrapper provides an additional abstraction layer between the domain object factory and the business component.
2. (Original) The framework of claim 1 wherein the database wrapper further comprises a database wrapper interface in communication with the business component and a database wrapper implementation implementing the domain object factory.
3. (Original) The framework of claim 2 wherein the domain object factory further comprises a domain object interface in communication with the database wrapper and a domain object factory implementation implementing the domain object.
4. (Original) The framework of claim 3 wherein the domain object further comprises a domain object interface in communication with the domain object factory and a domain object implementation retrieving data from a datastore.
5. (Original) The framework of claim 4 wherein the domain object interface further comprises a transient data converter for converting the domain object from a persistent state to a transient state.
6. (Original) The framework of claim 5 wherein the datastore is a relational database.
7. (Original) The framework of claim 5 wherein the datastore is an object database.
8. (Original) The framework of claim 5 wherein the datastore is accessed remotely.
9. (Previously Presented) A method for isolating a business component from specific implementations of a datastore, comprising:
   (a) interfacing a database wrapper to a business component;
   (b) implementing the database wrapper;

(c) interfacing a domain object factory to the database wrapper;

(d) implementing the domain object factory;

(e) interfacing a domain object to the domain object factory; and

(f) implementing the domain object to retrieve data from a datastore, wherein the database wrapper provides an additional abstraction layer between the domain object factory and the business component.

10. (Original) The method of claim 9 further comprising converting data retrieved from the datastore from a persistent state to a transient state.

11. (Original) The method of claim 10 wherein the datastore is a relational database.

12. (Original) The method of claim 10 wherein the datastore is an object database.

13. (Original) The method of claim 10 wherein the datastore is accessed remotely.

14. (Previously Presented) A method for isolating a business component from specific implementations of a datastore, comprising:

D1

(a) supplying a database wrapper;

(b) using the database wrapper to begin a database session;

(c) using the database wrapper to obtain a domain object factory;

(d) using the domain object factory to create a domain object;

(e) converting the domain object from a persistent state to a transient state;

(f) ending the database session; and

(g) returning the domain object to the business component, wherein the database wrapper provides an additional abstraction layer between the domain object factory and the business component.